

Ćwiczenie 1

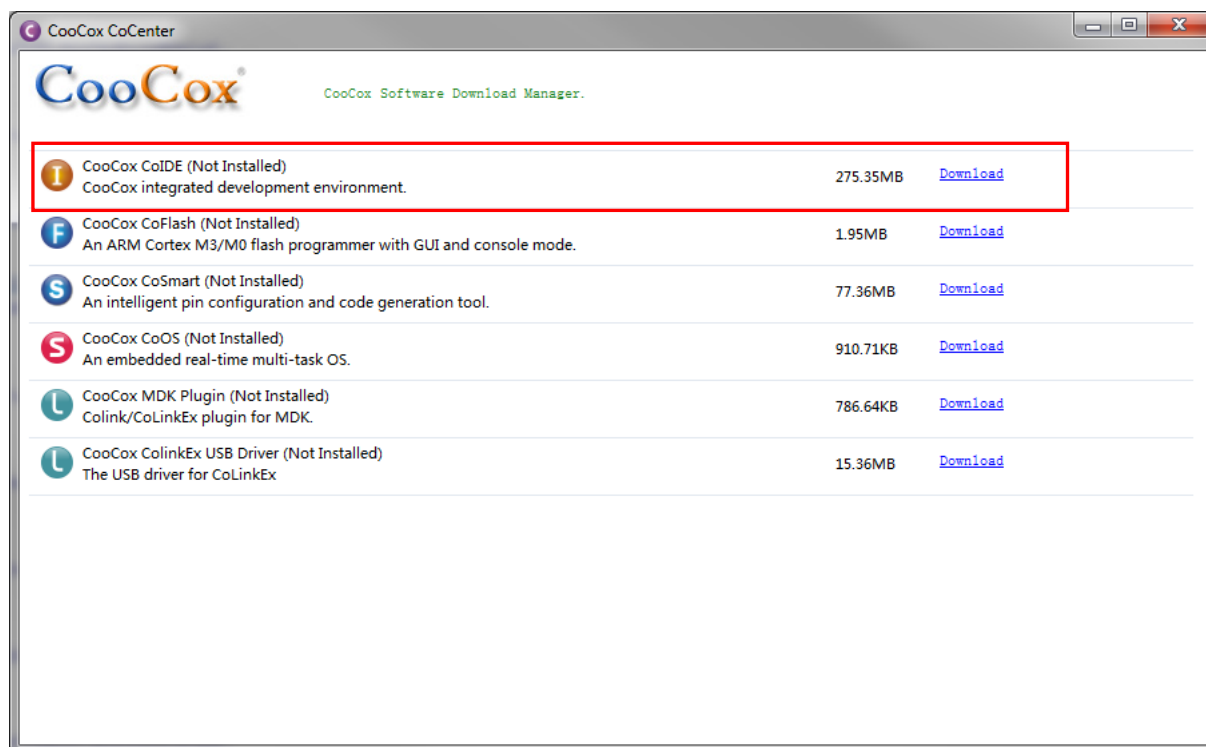
Wprowadzenie, zapoznanie ze środowiskiem programistycznym oraz zestawem uruchomieniowym. Kompilacja i programowanie układu.

1. Wstęp

Przed rozpoczęciem pracy z mikrokontrolerami z rodziny STM32F4 firmy ST Microelectronics należy pobrać i zainstalować odpowiednie narzędzia. Na zajęciach korzystać będziemy z dostępnego bezpłatnie pakietu CoCoX. Na pakiet składa się dostosowana wersja środowiska programistycznego Eclipse, które współpracuje z kompilatorem ARM-GCC, a także dodatki, takie jak sterowniki popularnych urządzeń peryferyjnych oraz darmowy, otwarty system czasu rzeczywistego CoOS. Środowisko umożliwia również pełne debugowanie projektu z wykorzystaniem wielu typów adapterów JTAG.

2. Instalacja składników CoCoX

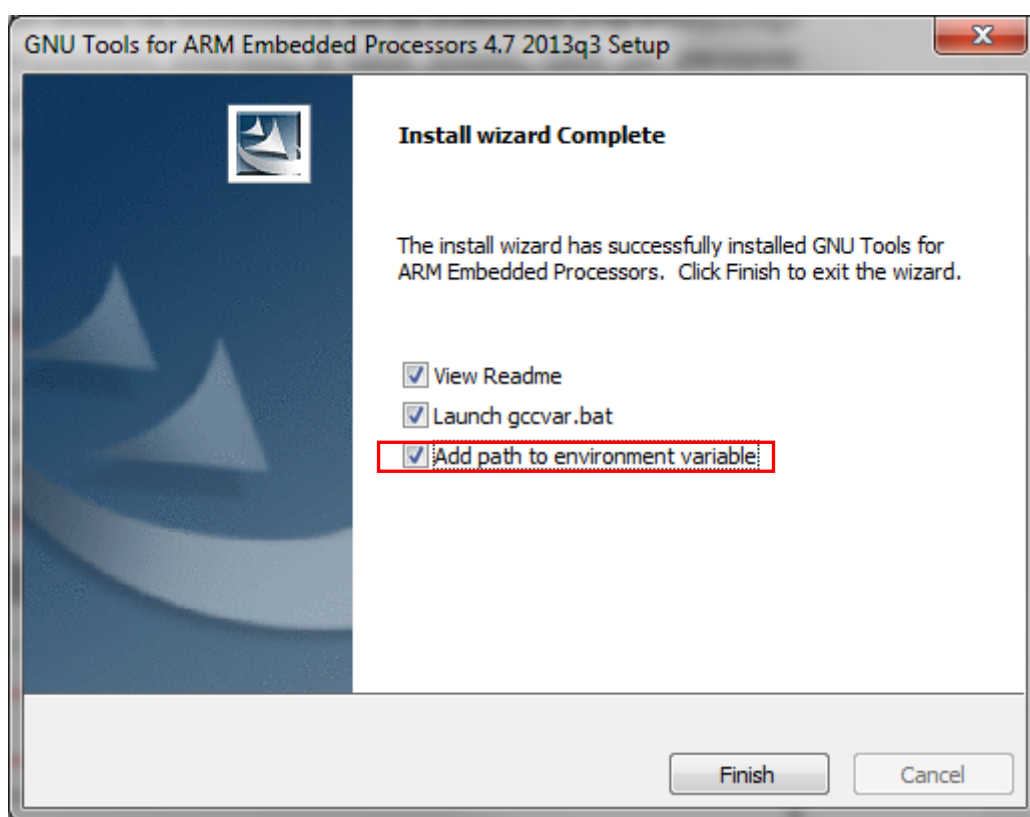
Aby zainstalować składniki pakietu należy ze strony [HTTP://WWW.COCCOX.ORG/COCENTER.HTM](http://www.cocox.org/CoCenter.htm) pobrać program CoCenter, po czym wybrać składniki do zainstalowania. Aby pobrać program należy założyć konto na stronie. Po dokonaniu wyboru pobrany i uruchomiony zostanie plik instalacyjny. Na początek wystarczy zainstalować CoIDE (patrz rysunek poniżej):



Z poziomu programu CoCenter można uruchamiać, dodawać, usuwać oraz aktualizować składniki pakietu zgodnie z potrzebami.

3. Instalacja kompilatora ARM-GCC

Ze strony [HTTPS://LAUNCHPAD.NET/GCC-ARM-EMBEDDED/+DOWNLOAD](https://launchpad.net/gcc-arm-embedded/+download) należy pobrać kompilator ARM-GCC – dostępna jest wersja z instalatorem i zalecane jest jej użycie. Podczas instalacji należy wybrać opcję dodania odpowiednich ścieżek do plików systemowych (patrz rysunek poniżej):



4. Instalacja sterownika ST-LINK v2

Aby mieć możliwość korzystania z adaptera JTAG zamontowanego na płytce prototypowej, która wykorzystywana będzie na zajęciach należy zainstalować odpowiednie sterowniki. W tym celu, należy wejść na stronę pod adresem: [HTTP://WWW.ST.COM/INTERNET/EVALBOARD/PRODUCT/251168.JSP](http://www.st.com/Internet/EVALBOARD/PRODUCT/251168.JSP), wybrać zakładkę 'Design Resources' i pobrać oraz zainstalować plik:

[Home](#)
[Products](#)
[Applications](#)
[Support](#)
[Sample & Buy](#)
[About](#)
[Contact](#)
[My ST Login](#)
[Parametric Search](#)

[Home](#) > [Tools and Software](#) > [Development Tool Hardware](#) > [Development Tool Hardware for MCUs](#) > [Development Tool Hardware for STM8 MCUs](#) > [Debug Hardware for STM8 MCUs](#) > [ST-LINK/V2](#)

[Save to MyST](#)
[Share](#)
[Print](#)

[Quick View](#)
[Design Resources](#)
[Sample & Buy](#)
[All](#)

ST-LINK/V2

ST-LINK/V2 in-circuit debugger/programmer for STM8 and STM32

● Active

Quick Links [Product Specifications](#)

Technical Documentation

Product Specifications		
Description	Version	Size
DB1275: ST-LINK/V2 in-circuit debugger/programmer for STM8 and STM32	3.0	94 KB

User Manual		
Description	Version	Size
UM1075: ST-LINK/V2 in-circuit debugger/programmer for STM8 and STM32	6.0	514 KB

Release Notes		
Description	Version	Size
RN0084: ST-LINK/V2 firmware upgrade	1.0	41 KB

Related Tools and Software

Related Tools and Software	
Part Number	Description
STSW-LINK003	ST-LINK/V2 USB driver for Windows 7, Vista and XP
STSW-LINK004	STM32 ST-LINK utility
STSW-LINK006	ST-LINK/V2 USB driver for Windows 8

Online Support

[Online Support](#)

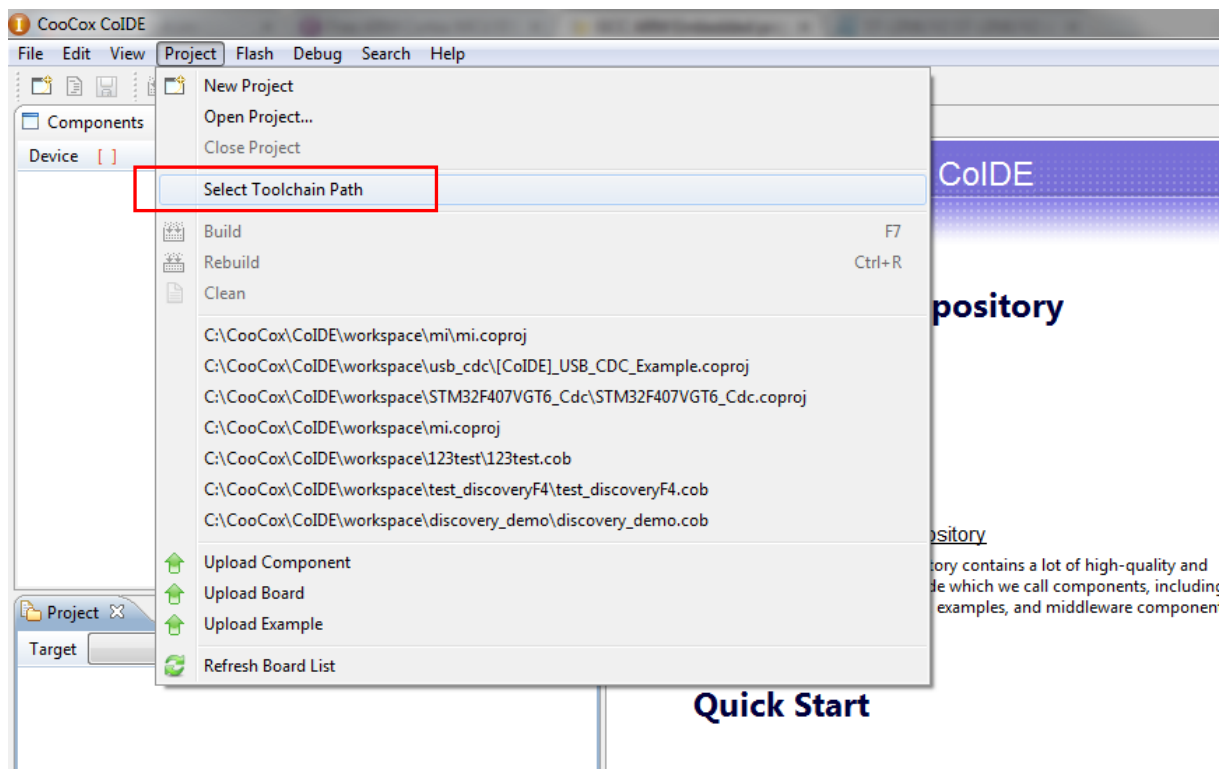
[FAQ](#)

[E2E Communities](#)

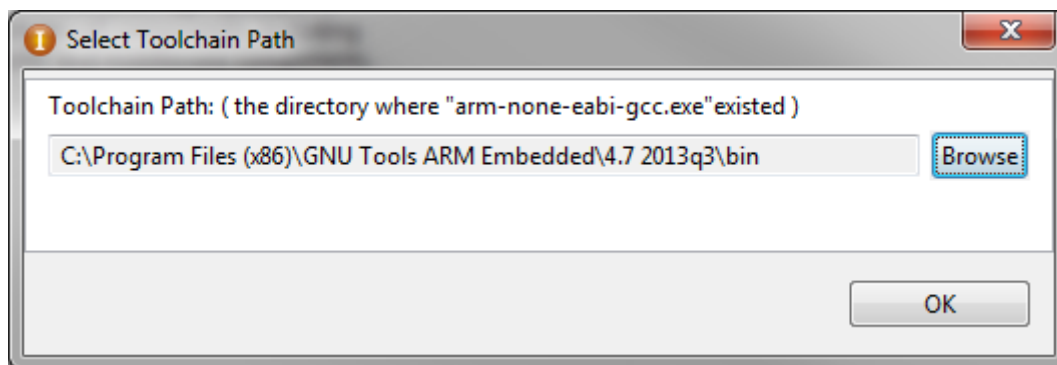
[Learning](#)

5. Konfiguracja CoIDE

Aby umożliwić współpracę CoIDE z kompilatorem, należy wskazać ścieżkę dostępu do plików wykonywalnych narzędzi ARM-GCC. W tym celu po uruchomieniu CoIDE należy w górnym rozwijanym menu wybrać opcję Project -> Select Toolchain Path (patrz rysunek poniżej):



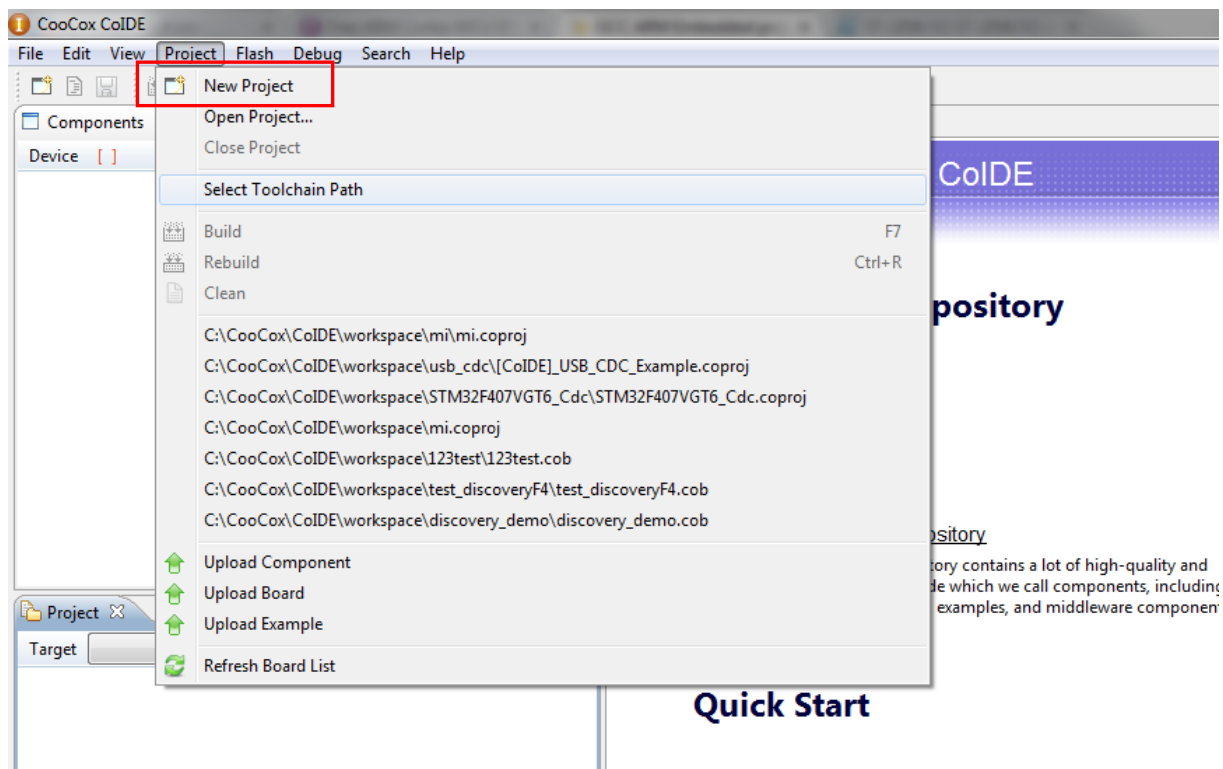
W okienku, które otworzy się po wybraniu opcji wpisujemy ścieżkę do katalogu ...**\bin** naszej instalacji ARM-GCC. W przypadku podaniu parametrów domyślnych ścieżka wyglądać może następująco:



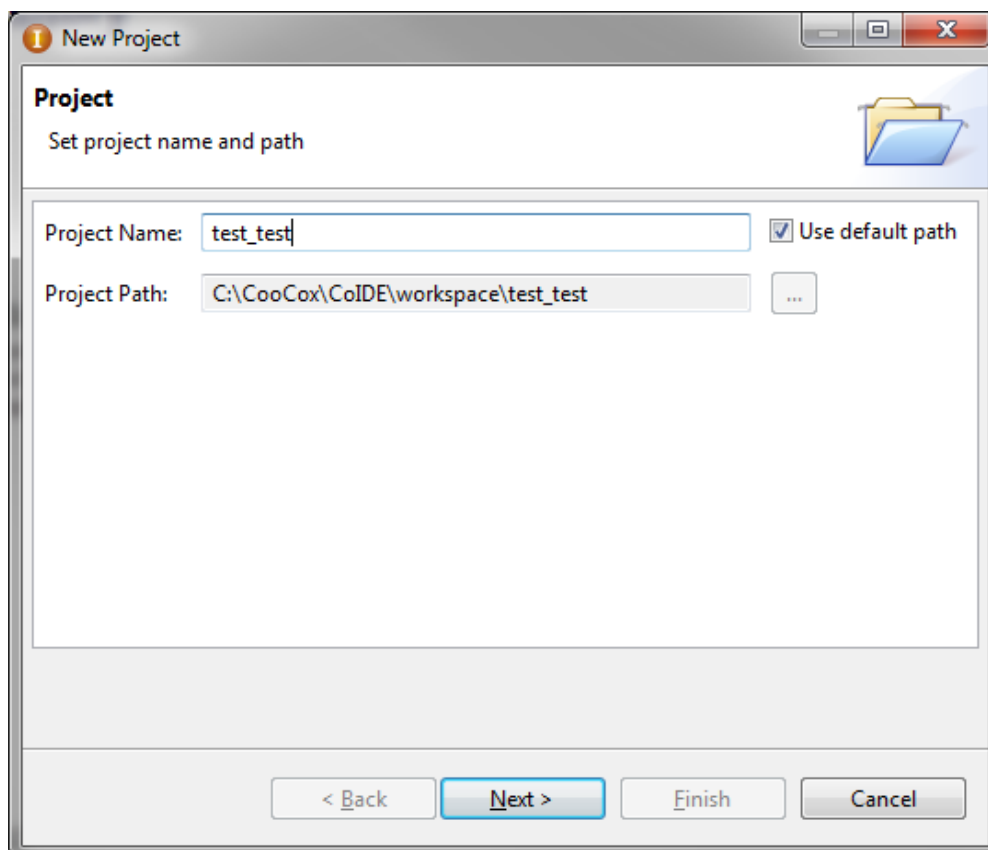
Jesteśmy już gotowi do utworzenia pierwszego projektu.

6. Tworzenie i konfiguracja projektu

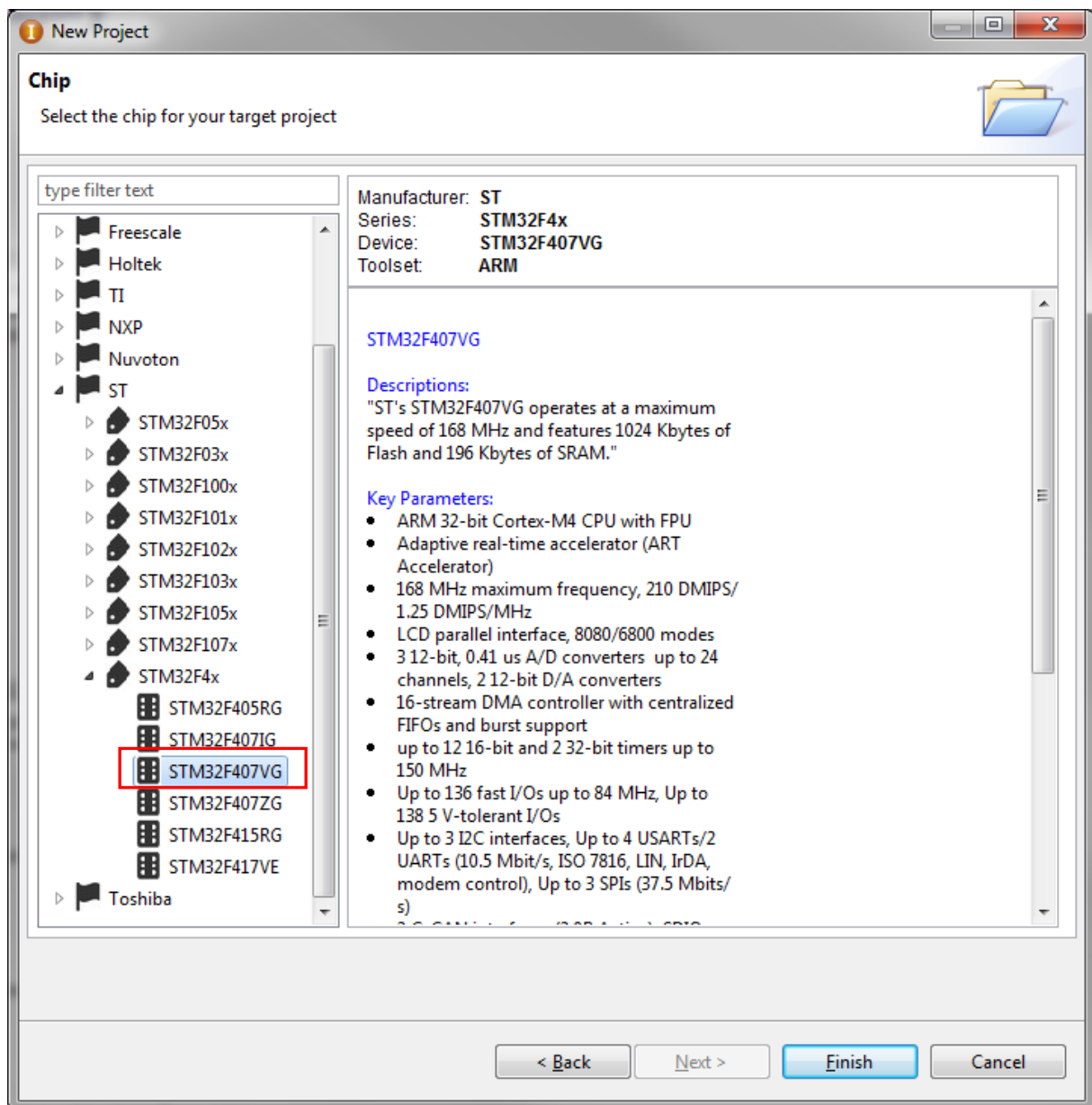
W menu 'Project' wybieramy opcję 'New Project' (patrz rysunek poniżej):



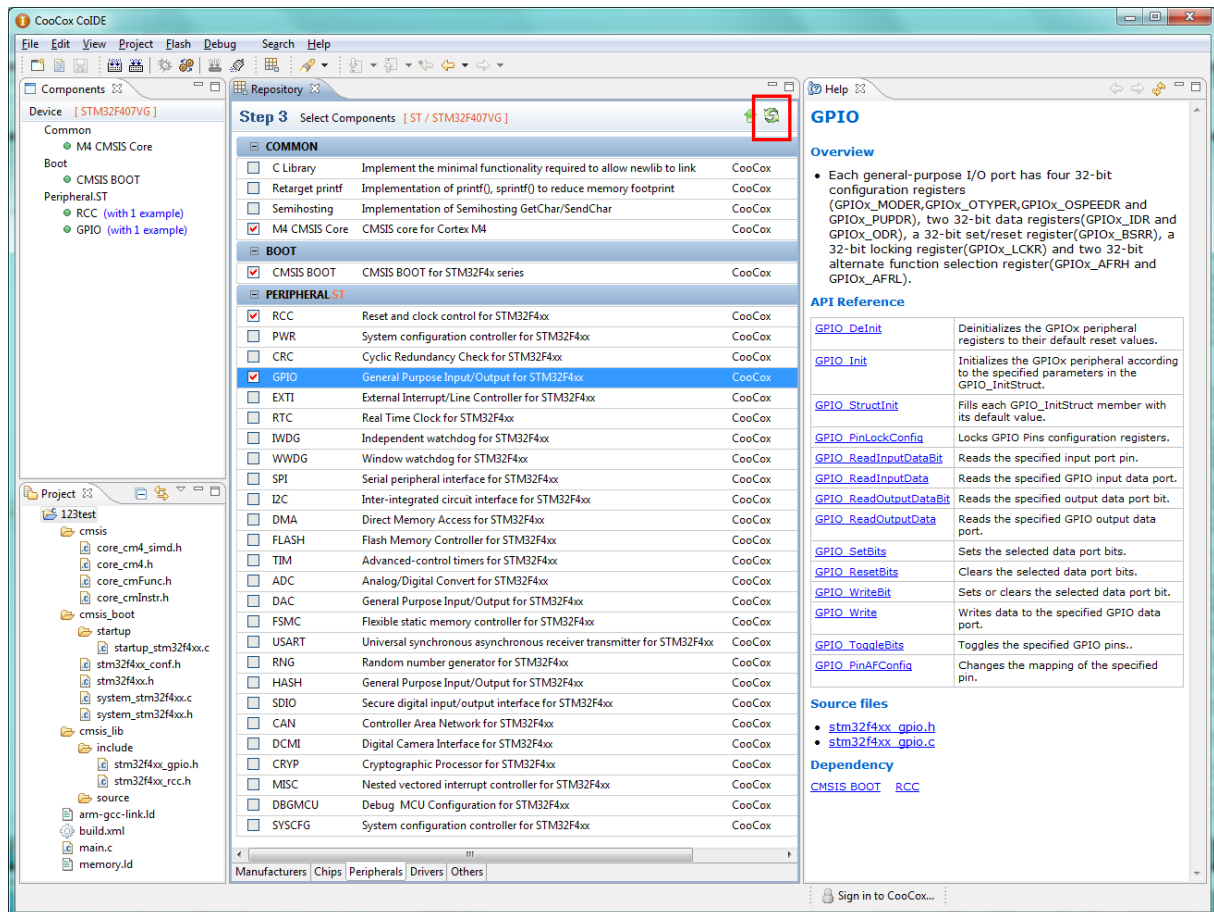
W oknie, które zostanie otwarte wpisujemy nazwę projektu i klikamy 'Finish':



Następnie wybieramy opcję 'Chip' i z listy wybieramy układ STM32F407VG



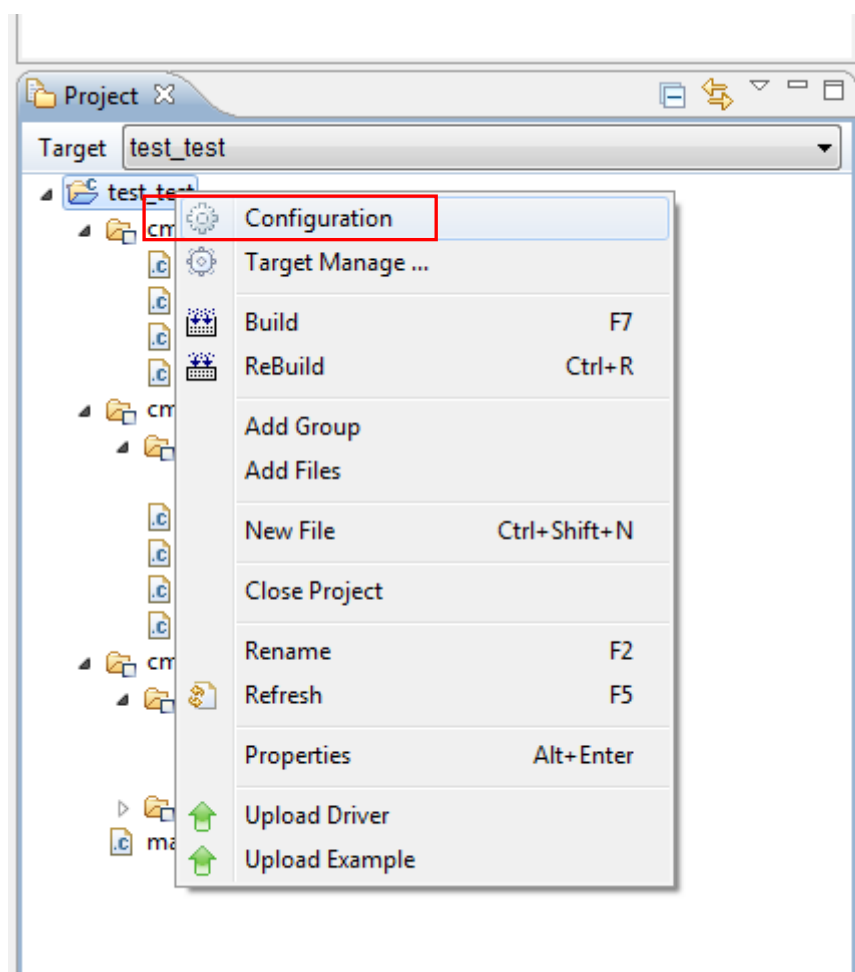
W kolejnym oknie zaznaczamy, które z plików sterowników układów peryferyjnych mikrokontrolera mają zostać dodane do projektu. Należy zaznaczyć opcje jak na rysunku poniżej. Spowoduje to dodanie plików rozruchowych dla rdzenia ARM Cortex-M4, oraz sterowników układów zarządzających zegarem systemowym (RCC) oraz portami wyjściowymi mikrokontrolera (GPIO):



Sterowniki postać gotowych do wykorzystania funkcji. Proszę zwrócić uwagę na fakt, iż po zaznaczeniu sterownika w oknie po prawej pojawia się dokumentacja funkcji z nim związanych. Z widocznych tu plików pomocy można w każdej chwili skorzystać.

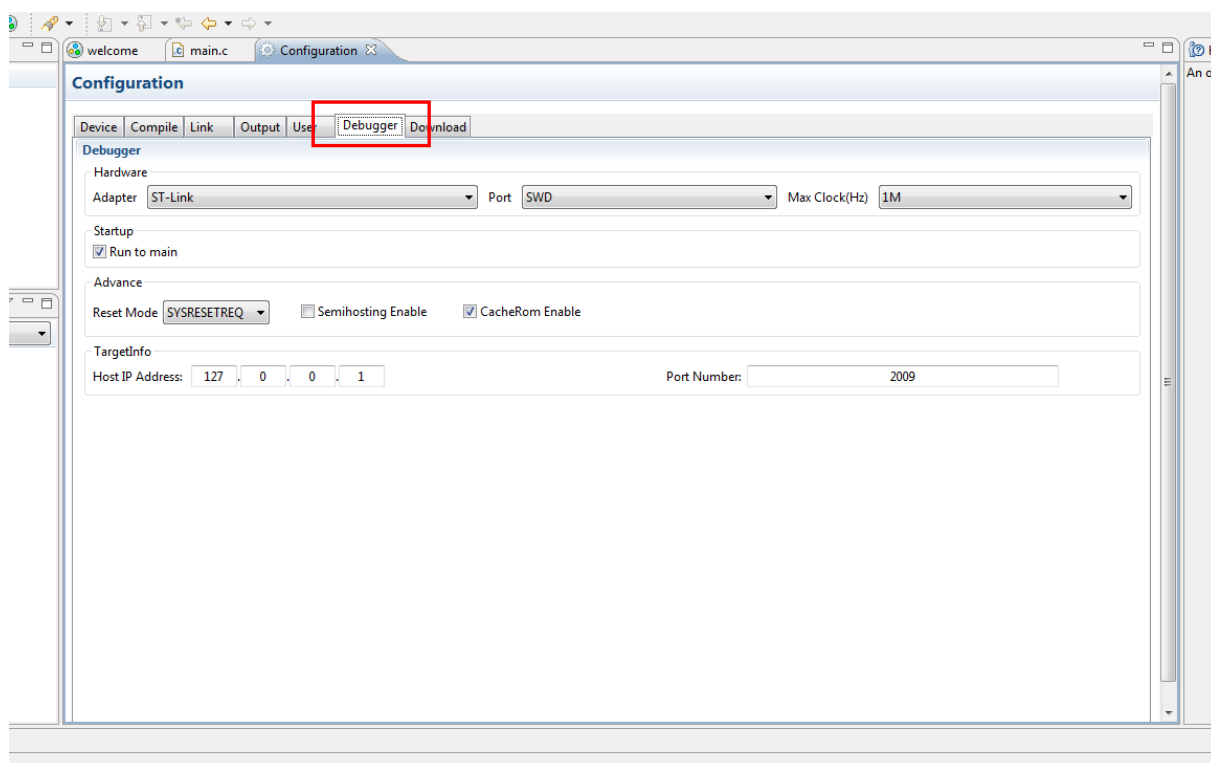
UWAGA: Jeśli nie wszystkie sterowniki peryferiów wypisane są jak w oknie powyżej, listę należy odświeżyć (przyciskiem oznaczonym na powyższym rysunku) oraz pobrać ewentualne brakujące pliki.

W następnej kolejności klikamy prawym przyciskiem myszy na nazwę projektu po lewej stronie i wybieramy opcję 'Configuration', aby wywołać okno ustawień projektu:



Proszę zwrócić uwagę na inne opcje dostępne w tym oknie, na przykład ustawienia optymalizacji kodu generowanego przez kompilator.

Aby umożliwić przesłanie plików wynikowych do mikrokontrolera gdzie zostaną wykonane, niezbędne jest prawidłowe skonfigurowanie adaptera JTAG, będącego integralną częścią wykorzystywanej na zajęciach płytki prototypowej. W tym celu wybieramy zakładkę 'Debugger' ekranu opcji i upewniamy się, że ustawienia są takie jak na rysunku poniżej:



Następnie, zawartość plików `system_stm32f4xx.c` oraz `main.c` podmieniać należy na zawartość z plików dostarczonych przez prowadzącego zajęcia. Dzięki podmianie pierwszego z plików zmienia się konfiguracja zegara systemowego mikrokontrolera z ustawień domyślnych na ustawienia prawidłowe dla płytki prototypowej wykorzystywanej na zajęciach. Zawartość pliku wygenerować można za pomocą narzędzia *STM32F4XX system clock configuration tool*, dostępnego jako załącznik do kursu. Do uruchomienia narzędzia wymagane jest zainstalowanie programu Microsoft Excel oraz włączenie obsługi makr. Do kursu dołączony jest również wygenerowany za pomocą narzędzia plik zawierający prawidłowe ustawienia. Zawartość pliku `main.c` po podmianie to prosty, przykładowy program sterujący diodami LED.

Dodatkowo, w pliku `stm32f4xx.h` należy zamienić linijkę:

```
#define HSE_VALUE ((uint32_t)25000000) /*!< Value of the External oscillator in Hz */
```

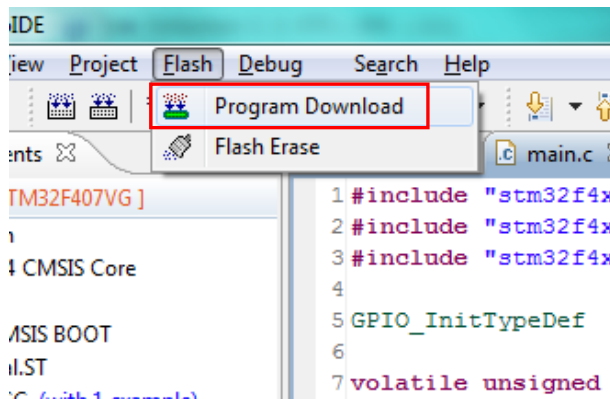
na:

```
#define HSE_VALUE ((uint32_t)8000000) /*!< Value of the External oscillator in Hz */
```

Jest to konieczne, gdyż częstotliwość rezonatora kwarcowego zamontowanego na płytce wynosi 8 MHz, podczas gdy według domyślnie generowanych w plikach ustawień jest to 25 MHz.

Jeśli wszystkie kroki wykonane zostały poprawnie, po wciśnięciu klawisza F7 projekt powinien się poprawnie skompilować.

Jeśli adapter JTAG został poprawnie skonfigurowany, po wybraniu z menu 'Flash' opcji 'Program Download' (patrz rysunek poniżej) plik wynikowy zostanie przesłany do mikrokontrolera i wykonany.



Wciśnięcie kombinacji klawiszy Ctrl+F5, lub wybranie opcji 'Debug' z menu 'Debug' umożliwia śledzenie działania programu krok po kroku. Możliwe jest również ustawianie pułapek (poprzez dwukrotne kliknięcie lewym klawiszem myszki z lewej strony interesującej nas linii programu, ew. za pomocą kombinacji klawiszy Ctrl+Shift+B) oraz podglądanie aktualnego stanu zmiennych podczas pracy mikrokontrolera.